CLAIMS

1. A two-way interactive terminal system comprising:

a host data processor;

a central data store controlled by the host processor for storing a central data base, said central data base comprising a plurality of data items, each of said data items having a preassigned priority value; and

at least one local terminal subsystem connectable to the host processor for accessing data items stored in the central data base, said terminal subsystem comprising:

a local processor for controlling access to the host processor;

memory means controlled by the local processor and comprising a first memory area for storing data items accessed from the central data base;

means operable when the first memory area has no vacant space for storing a newly accessed data item to locate a data item in said first memory area for deletion based on the relative priority values of the data items; and

means responsive to said locating means to delete the data item located by the locating means to create space in the first memory area for the newly accessed item.

2. The terminal system of claim 1 wherein said memory means further comprises a second memory area for storing a directory entry for each of said data items stored in the first memory area, said directory entries specifying the memory address in said first memory area of each of the data items and being stored in said second memory area in sorted order according to the relative priority of the data items.

3. The terminal system of claim 2 wherein said local terminal subsystem further comprises sorting means for updating and sorting said directory entries whenever the contents of the first memory area are changed.

4. The terminal system of claim 2 wherein said locating means locates a data item to be deleted by accessing said second memory area.

5. The terminal subsystem of claim 1 further comprising a keyboard for inputting information to the local processor.

6. The terminal subsystem of claim 5 wherein said local processor separately activates a plurality of system features responsive to the actuation of predetermined keys of the keyboard.

7. The terminal system of claim 6 wherein the data items stored in the first memory area comprise a plurality of character screen commands and wherein said terminal subsystem further comprises a character generator, responsive to a character screen command from the local processor, for generating a character display screen for display on an associated television receiver.

8. The terminal system of claim 7 wherein said memory means further comprises a third memory area for storing a channel map containing a plurality of entries, the entries identifying for each of the system features a predetermined keyboard key and a predetermined character screen command, the character screen command being accessed by said local processor for display on the television receiver responsive to actuation of the predetermined key.

9. The terminal system of claim 7 wherein each of the character screen commands has a preassigned feature number corresponding to one of said plurality of system features.

10. The terminal system of claim 9 wherein if said first memory area is full and said local processor has activated a system feature, said locating means locates a data item in said first memory area for deletion having the lowest relative priority value and having a feature number different than the activated feature.

11. The terminal system of claim 9 wherein said character generator, responsive to at least one of said character screen commands, generates a character display screen requiring a keyboard response, and wherein said at least one of said character screen commands includes a pointer specifying another character screen

command to be accessed by the local processor responsive to the keyboard response.

12. The terminal system of claim 11 wherein when the feature of said at least one character screen command has been activated, if no space is available in said first memory area for storing a newly accessed item and if all of the character screens stored in said first memory area contain a feature number which is the same as the feature number of said at least one character screen command, said locating means locates a data item in said first memory area, other than said another character screen command, having the lowest relative priority value.

13. The terminal system of claim 9 wherein a cannibal bit is associated with each of said character screen commands, and wherein if no space is available in said first memory area for storing a newly accessed item and the local processor has activated a system feature, the local processor allows deletion of a character screen command having a feature number corresponding to the activated feature if the cannibal bit of the newly accessed item is 1 and the local processor inhibits deletion of a character screen command having a feature number corresponding to the activated feature if the cannibal bit of the newly accessed item is 0.

14. In a two-way interactive terminal system comprising a host data processor, a central data store controlled by the host processor for storing a central data base, said data base comprising a plurality of data items, each of the data items having a preassigned priority number, at least one local terminal subsystem connectable to the host processor for accessing data items stored in the central data base, said terminal subsystem comprising a local processor for controlling access to the host processor and memory means controlled by the local processor comprising a first memory area for storing data items accessed from the central data base, a method for controlling the storage of data items in said first memory area comprising the steps of:

determining whether said first memory means contains vacant space for storing a data item accessed from said central data base;

storing said newly accessed data item in said first memory area if vacant space exists in the first memory area; and

locating a data item in said first memory area for deletion based on the relative priority values of the data items and replacing the located data item with the newly accessed data item when no vacant space exists in the first memory area.

add a2/   Add D1>